

this handout and some of the references quoted at
www.unice.fr/scheer/

ONE-CHANNEL TRANSLATION

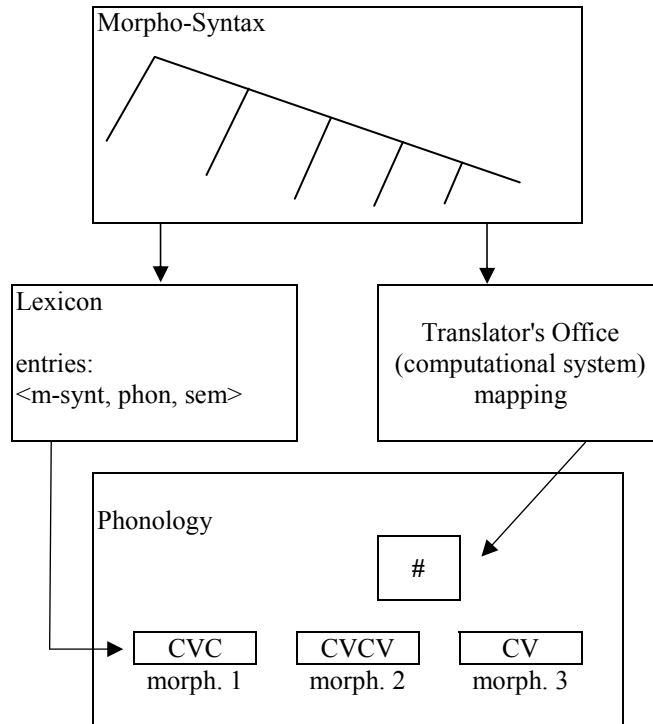
PREQUEL OF ARBITRARINESS IN POST-PHONOLOGICAL SPELL-OUT

1. Scope

- (1) What this talk is about
 - a. the representational means of morpho-syntax to talk to phonology
 - b. the way representational carriers of morpho-syntactic information (hash-marks, prosodic constituency etc.) come into being, i.e. the translational process
 - c. candidates:
 1. computational translation
 2. lexical translation (dictionary)
 - d. the maybe underestimated complexity of modular transitions
 1. PF - a hermaphrodite modular object of wonder
 2. how many phonetics?
- (2) What this talk is not about
 - a. the derivational means for morpho-syntax to talk to phonology
cycles (70s, 80), domains (GP 90s), phases (current syntax since 00s)
 - b. the nature of the output of translation
(although we get to this at some point)
structuralist juncture phonemes, SPE-type diacritics (hash marks), autosegmental prosodic constituency (80s to present), non-diacritic items (syllabic space, GP 90s to present).

2. How translation has always worked: Two Channels

- (3) Translation in generative interface thinking
Mixed lexical and computational translation:
Two Channels



- (4) lexical translation
- concerns **morphemic information** of the output of morpho-syntactic computation
 - morphemic information is matched with lexical entries, which compete for insertion.
 - lexical entries are made of three types of information
 - morpho-syntactic
 - phonological
 - semantic
 - look-up
translation is through a lexical access that works like a multilingual dictionary:
you know the item of one language and want to know what their equivalence is in another language.
 - example
 - <number = plural>
 - <person = 3>
 - <verb class = X>

==> morpheme injected into phonology: -ou

- (5) computational translation
 - a. concerns **non-morphemic information** (or boundary information) of the output of morpho-syntactic computation
 - b. the items inserted into the linear string
 - 1. are not stored in the lexicon (long-term memory)
 - 2. exist only during online computation (working memory)
 - 3. are created through a computational action that transforms a (morpho-syntactic) input into a (phonological) output according to a fixed set of computational instructions (that is stored in long-term memory).
E.g. the SPE-algorithm that inserts hash marks at the edges of every major category (N, V, A) and of every projection thereof (NP, VP etc.).

3. Core properties of translation (in general)

- (6) partial homology
Jackendoff (2002)
 - a. translation is selective
only a subset of the properties of the sending module is made available to the receiving module.

"Correspondence rules perform complex negotiations between two partly incompatible spaces of distinctions, in which only certain parts of each are 'visible' to the other." Jackendoff (1997:221)

"The overall architecture of grammar consists of a collection of generative components $G_1, \dots, G_n$ that create/ license structures $S_1, \dots, S_n$, plus a set of interfaces I_{jk} that constrain the relation between structures of type S_j and structures of type S_k . [...] Typically, an interface I_{jk} does not 'see' all of either S_j or S_k ; it attends only to certain aspects of them." Jackendoff (2002:123)
 - b. information bottleneck
the amount of structure that is visible for interface processors in a given module may be small or big, and this is unpredictable: the translational channel between two modules may have a more or less narrow "information bottleneck" (Jackendoff's 2002:229 term).
 - c. well supported in language:
morpho-syntax and melody are incommunicado in both ways
- (7) translation is arbitrary
Jackendoff (2002)
 - a. which pieces of the structure of the sending module are translated cannot be predicted.
 - b. well supported in language:
the mapping puzzle (Scheer 2011): all efforts at finding cross-linguistic patterns of translation have been by and large vain. That is, phonologists could not come up with natural classes of boundaries.

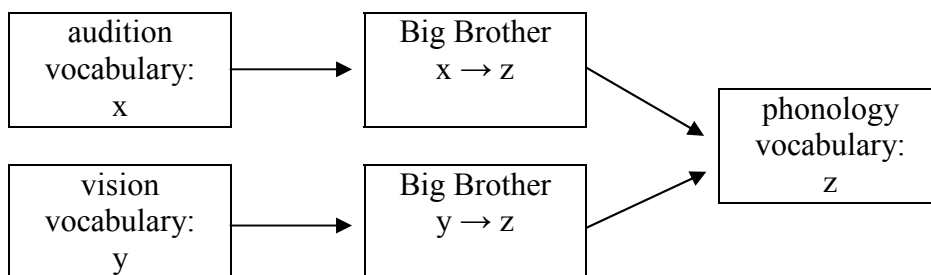
- (8) modules receive variable inputs, but produce a uniform output
 - a. many-to-one
 - modules may draw on information that comes from a range of other modules
 - 1. example: in perception, phonology is fed at least by acoustic-phonetic and visual information.
==> McGurk effect (McGurk & MacDonald 1976, Ingleby & Azra 2003)
 - 2. the circuitry of visual stimuli that reach grammatical processing appears to be different from auditory stimuli, but processed by the auditory cortex (Calvert & Cambell 2003).
 - 3. complementary distribution
interestingly, the McGurk input into the phonological module appears to be the complementary set of what morpho-syntax can provide: melodic primes.
- (9) one-to-many
 - a. the output of a given module may be used as the input to a range of other modules
 - audition
 - provides information for a number of very different modules: sound is processed by
 - all-purpose audition (e.g. the perception of sound that is produced by animals)
 - voice recognition (the identification of humans according to their voice)
 - auditory affect perception (emotion detector)
 - perception of linguistically relevant phonetic material
 - b. consequence
 - variable input vocabularies that are all mutually unintelligible must be translated into the proprietary vocabulary of the receiving module.

4. Computational translation violates domain specificity

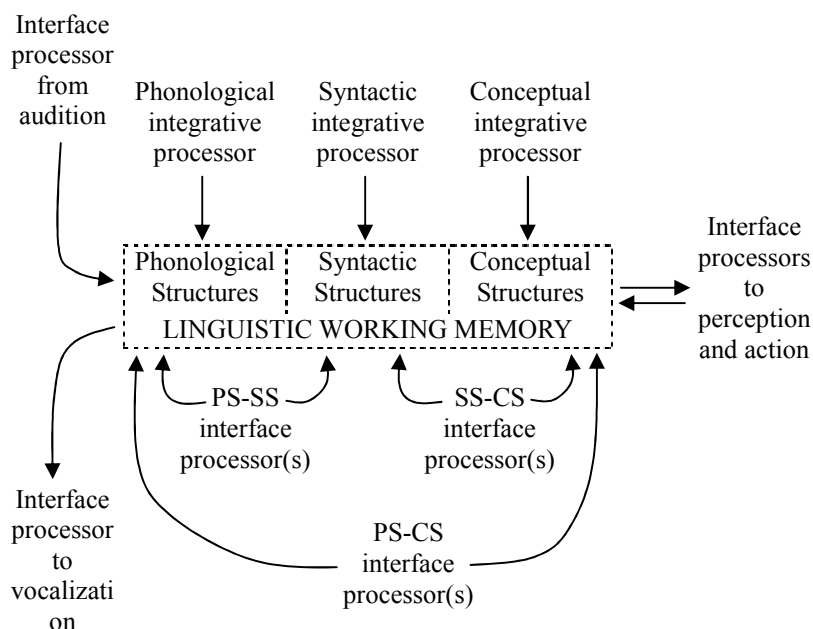
- (10) computational translation
 - a. translation of non-morphemic information has always been conceived of as computational
 - 1. readjustment rules (SPE)
 - 2. mapping rules (Prosodic Phonology)
 - 3. correspondence rules (Jackendoff)

all are a computation in its own right, i.e. distinct from either the sending or the receiving module.
 - b. Big Brother
 - translation by computation requires the Translator to have access to both the vocabulary of the sending and the vocabulary of the receiving module.
 - ==> violation of domain specificity
 - c. Jackendoff (2002:229) tries to discuss away this contradiction by resorting to the word "bi-domain specificity": interface modules are domain-specific like all others, but they are super-modules and therefore can be specific to two domains.
==> contradiction in terms
 - d. "a cynic might say therefore that the issue of modularity is dissolved. I would disagree" (Jackendoff 2002:229).
There is no further argument, though.

- (11) what is the status of computational devices that do translation?
- they can only be modules, since there is nothing in modular theory that carries out computation apart from modules.
==> but they cannot be modules because they violate domain specificity.
 - in Jackendoff's model (where modules are called processors):
 - inferential processors (Fodor's central systems)
 - integrative processors (Fodor's modules)
 - interface processors
 integrative processors are related by interface processors.
- (12) reduction of variable inputs to a uniform output
- no trouble for computational translation: on their input side, modules have a Big Brother for each different vocabulary that they are fed with.
 - example [audition, vision] → phonology



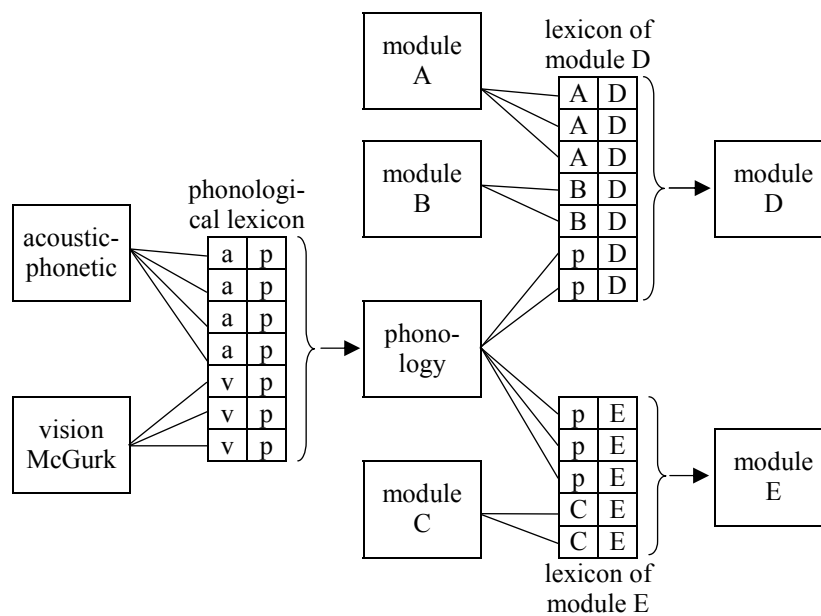
- (13) modular structure of language according to Jackendoff
(reproduction of the chart from Jackendoff 2002: 199)



5. One-Channel Translation

- (14) alternative to computational translation
 - a. all translation is only lexical
 - b. both morphemic and non-morphemic information is translated through a lexical access
 - c. all phonological material originates in the lexicon
 - d. what about the workings of translation in other cognitive functions? Is this currently done by computational mechanisms?
- (15) translation bears the signs of **lexical** activity
 - a. arbitrary relations of an input and an output
[as in dictionaries]
 - b. refusal to obey cross-linguistic lawful behaviour
[mapping puzzle]
- (16) other arguments for lexical translation
 - a. Big Brothers violate domain specificity
 - b. economy / Occam's Razor: lexical translation uses the resources of modular theory that are needed anyway.
Modularity knows only
 - 1. modules
 - 2. lexica
 - 3. eventually central systems
- (17) reduction of variable inputs to a uniform output
 - a. instead of having a number of Big Brothers, modules have a proprietary Lexicon on their input side.
 - b. this Lexicon has variable inputs (i.e. written in the distinct vocabularies of the different inputs), but a uniform output, i.e. only phonological vocabulary.
 - c. in this perspective, lexical entries are pairs of arbitrarily associated items which belong to two different domains.

(18) intermodular communication with lexical translation



6. Benefits: arbitrariness and its lexical limitation

(19) well-known problem of computational translation: all-powerfulness

- a. Jackendoff defends all-powerful translation against the criticism of overgeneration, i.e. the fact that unconstrained transmission of information allows for the description of existing as much as non-existing interface activity.

"correspondence rules are conceptually necessary in order to mediate between phonology, syntax, and meaning. It is an unwarranted assumption that they are to be minimised and that all expressive power lies in the generative components. [...] In other words, correspondence rules, like syntactic and phonological rules, must be constrained so as to be learnable. Thus their presence in the architecture does not change the basic nature of the theoretical enterprise." Jackendoff (1997:40)

- b. Jackendoff conceives of correspondence rules in the same way as of phonological or syntactic processes: translation is modular computation.
- c. given the modular status of translation, the only restriction that Jackendoff admits is learnability.

(20) lexical translation constrains translation: anything is not possible

- a. anything that reaches phonology must originate in the lexicon.
Hence boundary information must qualify for being stored in the lexicon
- b. consequence: no diacritics
the output of the translation of boundary information are necessarily pieces of the proprietary phonological vocabulary: only this kind of vocabulary can be stored in the lexicon.
- c. boundary information was always translated into diacritics:
+, #, ω, φ etc.
- d. diacritics violate modularity anyway
since they do not belong to the phonological vocabulary and hence cannot be parsed by phonological computation.
- e. explanation of the ban on diacritics
diacritics are out of business because they cannot be stored in the lexicon.

- f. Direct Interface (Scheer 2008, 2012)
its heart: being a non-diacritic theory of the interface
- (21) morpho-syntax has no bearing on phonological computation
[a personal note]
 - a. basic observation
morpho-syntax NEVER alters phonological computation: computational instructions cannot be its output.
 - b. there is nothing wrong with that in principle, though, if translation is computational: a computational instruction could be added to phonological instructions.
 - c. this is what I entertained at some point: Government and Licensing (which are computational actions) were carriers of morpho-syntactic information.
 - d. since computation is necessarily distinct from the lexicon, it cannot be stored in the lexicon and hence cannot be the output of translation.
- (22) lexicon = linear identity of boundary information I
 - a. observation
morpho-syntax has no bearing on morpheme-internal phonology
 - b. the linear input string to phonology is made of pieces that represent morphemic as well as non-morphemic information.
 - c. since both have the same origin – the lexicon –
 - 1. boundary information must have exactly the same linear identity as morphemes
 - 2. that is, it is pieced together by the linearization mechanism and just as morphemes must end up as identifiable pieces of the linear string.
 - d. if translation were computational,
it could well modify morpheme-internal properties:
 - 1. boundary information would not be concerned by linearization
 - 2. the linear string would exist prior to the impact of translation
- (23) lexicon = linear identity of boundary information II
 - a. linear vs. domain-based identity of boundary information
 - b. linear identity
 - 1. structuralist juncture phonemes: #, +
 - 2. SPE-type diacritics: #, +
 - c. non-linear replaced by domain-based identity in the early 80s
 - 1. prosodic constituency
 - 2. consequence of autosegmentalism
 - d. difference
 - 1. it does not make sense
to say that hash marks define domains: they identify a local spot in the linear string.
==> linear identity
 - 2. it does not make sense
to say that omegas etc. are preceded or followed by a morpheme: they dominate a number of morphemes.
==> non-linear identity

- e. if boundary information originates in the lexicon,
domain-based outputs of translation do not qualify since domains cannot exist in the lexicon: domains are always domains of a particular string of items, and these items exist only through online computation.
==> boundary information is necessarily linear.

7. What that means for post-phonological Spell-Out

- (24) phonetic interpretation
 - a. a GP notion: Harris & Lindsey (1995: 46ff)
 - b. translation of phonological into phonetic material is
 - 1. lexical (non-computational)
 - 2. hence arbitrary
 - c. this was the topic of my talk at the previous APAP, Scheer (2014)
- (25) observation about chunking
 - a. phonetics and phonology are sensitive to the same divisions of the linear string:
they are isomorphic with respect to chunks they are sensitive to.
 - b. these divisions originate in and are defined by morpho-syntax.
 - c. 1. phonological computation or
2. post-phonological spell-out
do not create any additional or genuine chunks
- (26) conclusion
 - a. chunk creation is a consequence of concatenation (Merge).
 - b. there is no chunk-creation in absence of concatenation.
 - c. phonology does not concatenate anything
==> hence phonological computation does not create any genuinely phonological chunks that could be transmitted to phonetics.
 - d. same conclusion reached by different means
Scheer (2013)
D'Alessandro & Scheer (in press)
- (27) Excursus: prosodic constituency
also defines chunks – but representationally
 - a. original version
(Selkirk 1984, Nespor & Vogel 1986)
 - 1. prosodic constituency is created by a computational operation distinct from Spell-Out: mapping rules (computational translation)
==> there are two different chunk-creating mechanisms
 - regular Spell-Out (Vocabulary Insertion) ==> derivational
 - mapping rules creating representational chunks ==> representational
 - 2. was explicitly non-isomorphic with morpho-syntactic structure
 - 3. non-isomorphism was introduced by mapping rules
 - 4. ==> creation of representational chunks through a parallel computation (computational translation)

- b. OTed version
constraint-based mapping (Align)
 - 1. prosodic constituency is created by *phonological* constraints that make direct reference to morpho-syntactic information
==> violation of modularity, back to SPE's morphological diacritics
 - 2. ==> creation of representational chunks without computational action
 - c. partially de-OTed version
Kratzer & Selkirk (2007)
 - 1. prosodic constituency put back under the control of Spell-Out: every phase creates a prosodic constituent.
 - 2. but only partial control of Spell-Out: prosodic constituency is of course much more complex than just one layer of constituents that corresponds to phases.
 - 3. unclear how the rest of the Prosodic Hierarchy is constructed: by some phonological means, but this means is not made explicit.
 - d. conclusion
prosodic constituency is the transformation of information that is created computationally (Spell-Out) into representations.
 - 1. why take the detour?
 - 2. can grammar afford to have two distinct chunk defining mechanisms, one derivational (Phases), the other representational (the latter being an avatar of the former)?
- (28) meaning of the vocabulary used
- a. Spell-Out
 - 1. is the operation that sends successive pieces of morpho-syntactic structure to phonology (and semantics)
 - 2. this is how chunks are created:
 - they always coincide with a morpho-syntactic division
 - they are the result of the piecemeal fire that phonology receives (and of nothing else)
 - b. Translation
 - 1. is the operation that transforms vocabulary items of one set into items of another set.
 - 2. is only lexical (i.e. through a lexical access, dictionary type)
 - 3. at the upper interface this is called Vocabulary Insertion
 - c. Spell-Out and Translation
are entirely distinct operations: one may occur without the other.
 - d. no Spell-Out at the lower interface
 - 1. or rather, Translation is a necessary property of intermodular communication: it occurs at all intermodular transitions.
 - 2. Spell-Out
on the other hand is a plug-in: it may or may not accompany Translation when modules communicate.
 - 3. since it is a consequence of concatenation and since phonology does not concatenate anything, it does not occur at the lower interface.

(29) status of phonetics

is phonetics a module, i.e. a piece grammar?

a. diagnostic #1

there must be a genuine symbolic vocabulary used by phonetic computation

c. diagnostic #2

phonetics should be sensitive to grammatical information from other modules

b. diagnostic #3

1. there must be some independence from the physical world

2. lying inside of grammar, phonetics is the result of grammaticalization of extra-grammatical, i.e. real world items.

3. we know from grammaticalized items – and more generally from categories of the cognitive system that represent real world items (categorization etc.) – that they mimic the real world and are inspired of it, but have their own life, i.e. impose their own logic that is distinct from the real world item.

Compare

- animate (biological, status of sponge) vs. grammatical (in Czech, God is +anim.)

- time (real world) vs. grammatical (tense)

- sonorant (physical: non-obstruent) vs. grammatical (phonological: [χ, ʁ])

(30) diagnostic #1

==> ??

a. is there is a symbolic vocabulary that phonetics runs on?

b. if so, what does it look like?

Is it articulatory or acoustic?

c. gestures (articulatory)?

Articulatory Phonology (Browman & Goldstein 1989 etc.)

d. but do articulatory items qualify?

There is evidence to the end that articulatory information is not linguistically meaningful

1. GP tradition of phonetic signature: Harris & Lindsey (1995), Backley (2011)

2. Hamann (2011)

e. what would be acoustic primes?

(31) diagnostic #2

==> yes

a. it is the case that phonetics is sensitive to chunks

b. these chunks are isomorphic with the ones phonology is sensitive to

c. that is, to the chunks that are created by morpho-syntactic Spell-Out

d. of course this fact is sometimes used to argue for phonology and phonetics being not really distinct, i.e. to blur modular contours.

(32) diagnostic #3

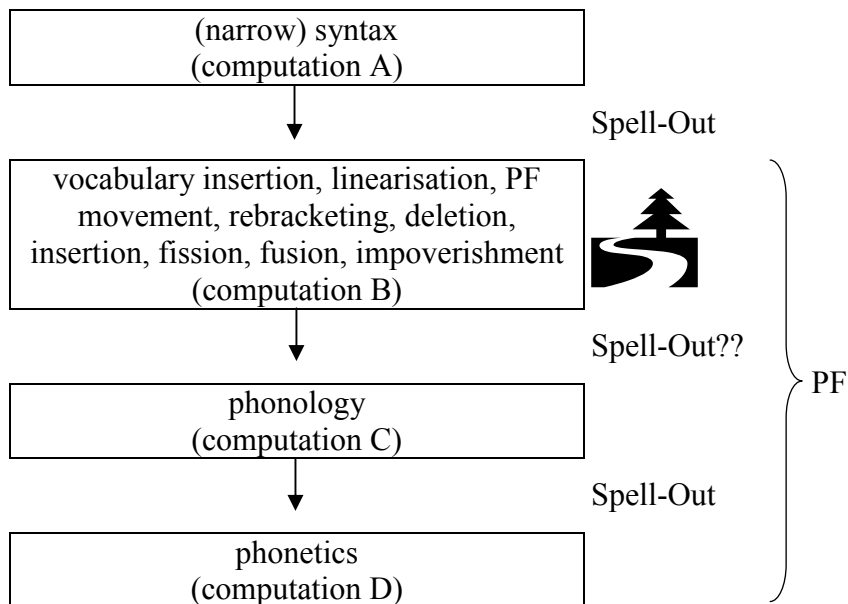
==> ??

is there a discrepancy between real world and phonetic items?

- (33) how many phonetics?
- are there maybe different phonetic systems, rather than just one?
 - which on the route
from grammar to non-grammar
gradually lose autonomy with respect to the real world
 - the central questions are
 - where exactly is the border line between grammar and non-grammar?
 - how is the transition managed?
 - recall SPE's systematic phonetics lying between phonology and "real" phonetics
 - based on gradience, which is gradually implemented
 - $[\pm\text{labial}] > [3,6 \text{ labial}]$
 - fought by GP (Harris 1996)

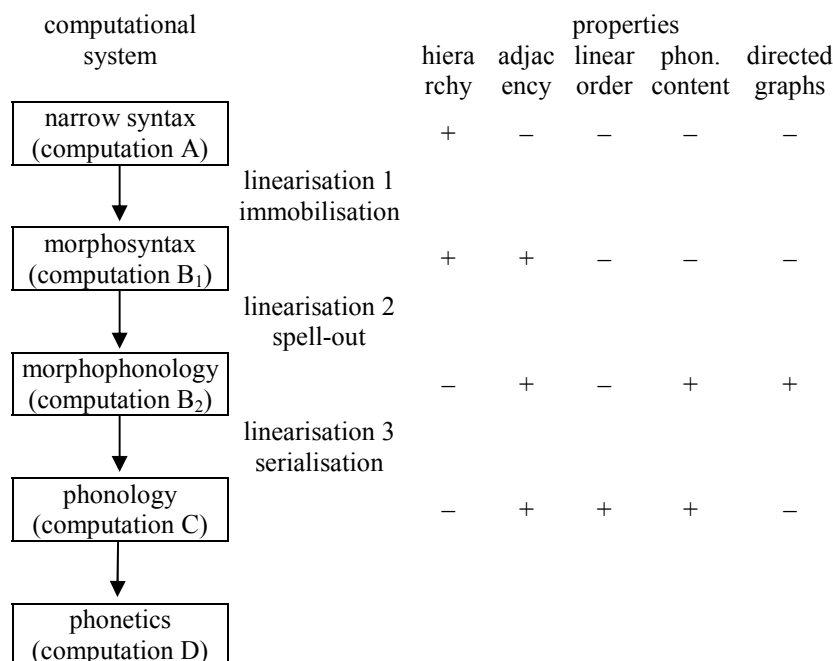
- (34) look at PF
- a hermaphrodite modular object of wonder inflated by minimalism
 - what's the internal structure of PF?
 - how many steps does the transition from Narrow Syntax to phonology take?
 - linearisation
still a mystery, much to be told, but little said.

- (35) PF
is a cover term for a number of serially ordered computational systems
==> this is what syntacticians mean when they talk about "PF"



(36) progressive 4-step linearisation using 5 criteria
according to Idsardi & Raimy (2013)

N.B.: this is only about linearisation, many other things go on "at PF"



References

- Backley, Phillip 2011. *An Introduction to Element Theory*. Edinburgh: Edinburgh University Press.
- Browman, Catherine P. & Louis M. Goldstein 1989. Articulatory gestures as phonological units. *Phonology* 6: 201-251.
- Calvert, Gemma & Ruth Cambell 2003. Reading Speech from Still and Moving Faces: The Neural Substrates of Visible Speech. *Journal of Cognitive Neuroscience* 15: 57-70.
- D'Alessandro, Roberta & Tobias Scheer in press. Modular PIC. *Linguistic Inquiry*.
- Hamann, Silke 2011. The Phonetics-Phonology Interface. *The Continuum Companion to Phonology*, edited by Nancy Kula, Bert Botma & Kuniya Nasukawa, 202-224. London: Continuum.
- Harris, John 1996. Phonological output is redundancy-free and fully interpretable. *Current trends in Phonology. Models and Methods*, edited by Jacques Durand & Bernard Laks, 305-332. Salford, Manchester: ESRI.
- Harris, John & Geoff Lindsey 1995. The elements of phonological representation. *Frontiers of Phonology*, edited by Jacques Durand & Francis Katamba, 34-79. Harlow, Essex: Longman. WEB.
- Idsardi, William & Eric Raimy 2013. Three types of linearization and the temporal aspects of speech. *Challenges to linearization*, edited by Ian Roberts & Theresa Biberauer, 31-56. Berlin: Mouton de Gruyter.
- Ingleby, Michael & Ali Azra 2003. Phonological Primes and McGurk Fusion. *Proceedings of the 15th International Congress of Phonetic Sciences*: 2609-2612. WEB.
- Kratzer, Angelika & Elisabeth Selkirk 2007. Phase theory and prosodic spellout: the case of verbs. *The Linguistic Review* 24: 93-135.

- McGurk, Harry & John MacDonald 1976. Hearing Lips and Seeing Voices. *Nature* 264: 746-748.
- Nespor, Marina & Irene Vogel 1986. *Prosodic Phonology*. Dordrecht: Foris.
- Scheer, Tobias 2008. Why the Prosodic Hierarchy is a diacritic and why the Interface must be Direct. *Sounds of Silence: Empty Elements in Syntax and Phonology*, edited by Jutta Hartmann, Veronika Hegedüs & Henk van Riemsdijk, 145-192. Amsterdam: Elsevier. WEB.
- Scheer, Tobias 2012. Direct Interface and One-Channel Translation. A Non-Diacritic Theory of the Morphosyntax-Phonology Interface. Vol.2 of *A Lateral Theory of phonology*. Berlin: de Gruyter.
- Scheer, Tobias 2013. Why phonology is flat: the role of concatenation and linearity. *Language Sciences* 39: 54-74.
- Scheer, Tobias 2014. Spell-Out, Post-Phonological. *Crossing Phonetics-Phonology Lines*, edited by Eugeniusz Cyran & Jolanta Szpyra-Kozłowska, 255-275. Newcastle upon Tyne: Cambridge Scholars.
- Selkirk, Elisabeth 1984. *Phonology and Syntax: The Relation between Sound and Structure*. Cambridge, Mass.: MIT Press.